Gaseous Leak Imaging by Raman Radiation Capture

W.T. Powers/EB22
205–544–3452
E-mail: wt.powers@msfc.nasa.gov

The period FY96 covers work that forms the conclusion of one contract increment and the majority of another contract increment.

One year ago a confocal telescope prototype had been constructed, and demonstration of proof of the concept was still underway. The goal was a simple 11×11 pixel image of a hydrogen leak inside a small test chamber.

Figures 86 and 87 display the layout of the confocal telescope breadboard. Figures 88 and 89 display data obtained using the breadboard. Figure 88 proves that it is possible to obtain a Raman scattering signal using a continuous wave (CW) probe laser.

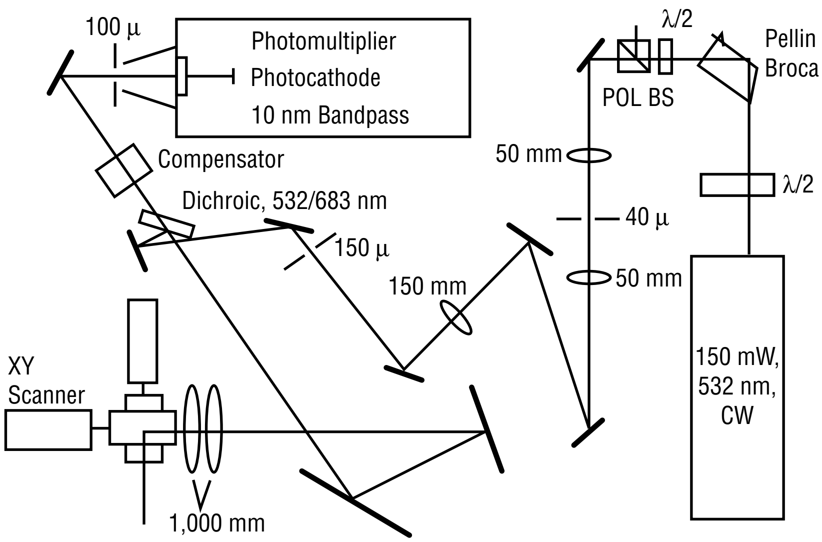

FIGURE 86.—Confocal telescope breadboard layout.

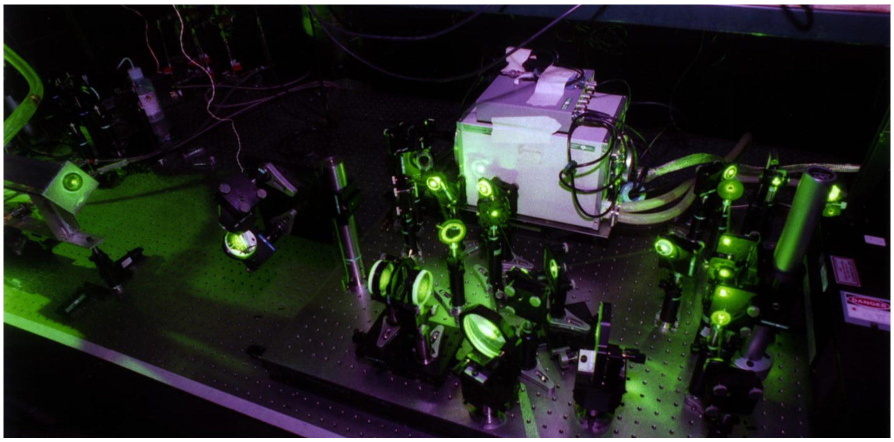

FIGURE 87.—Laboratory breadboard.

The background was aluminum at a distance of 70 cm (from the probe focus). Using the previous breadboard, it was impossible to obtain any signal even when using a pulsed laser and a 5 ns time gate. Figure 89 represents the first hydrogen leak image ever obtained using this new technology.

The confocal telescope breadboard can have its performance improved considerably by using a more elaborate narrow band filter. Use of a pulsed laser would greatly increase signal quality. The fact that data could be obtained using a CW laser opens the possibility of using a CW laser in a fully developed unit.

In March 1996 the project was refunded under a different program. The image of figure 89 was obtained by laboriously scanning the scene by hand and purging the chamber between pixels. The next step is to install a fast control system which can turn the confocal telescope breadboard into a real-time imaging system. The target imaging rate for the current phase is 300 pixels/sec.

The control system needed to drive the image system must be fast, must be built cheaply, and cannot use conventional software such as LabVIEW due to the performance requirements. A conventional PC hardware platform was long ago chosen for development to allow usage of lab computers. Extended DOS was chosen as the software platform to avoid 16-bit limitations, which are severe, and avoid dependence on slow and complex operating systems. The software effort is loosely categorized as an object oriented architecture, which allows maximum flexibility.

One software innovation which is mature is the cloning of 32-bit-protected mode-interrupt handlers written in C to real mode 16-bit mode, which converts each real-time interrupt function into a dual mode function. This avoids mode shifts upon an interrupt. This feature may be important as the software is migrated to a PC/104 platform.

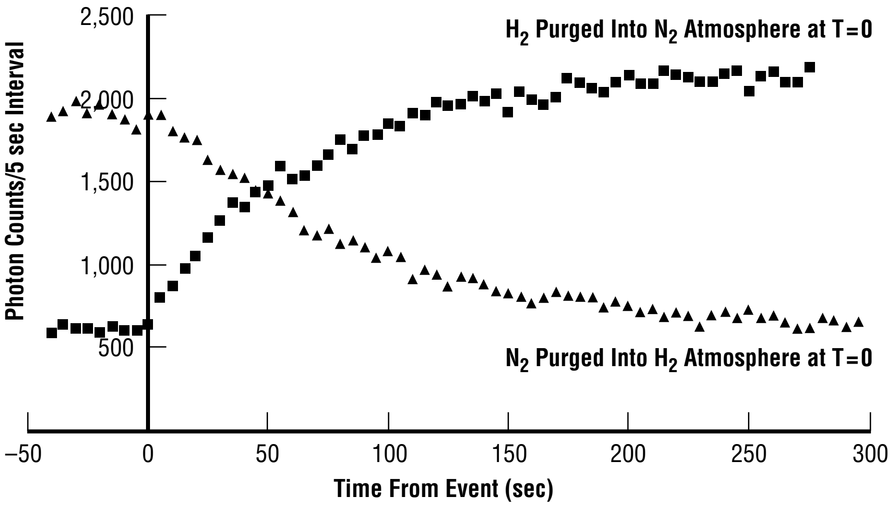

FIGURE 88.—Signal plus fluorscence versus time as chamber is filled and purged of H_2 content.

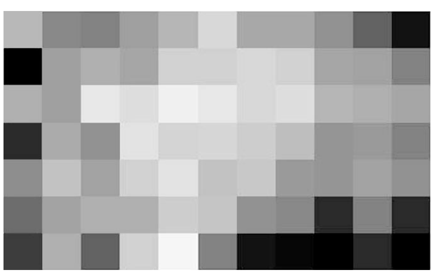

FIGURE 89.—Simple plume image 7×12.

Figures 90 and 91 display the software architecture which is being written. At present the software is being integrated with the circuit shown in figure 92 to give maximum imaging speed. The goal of the current phase is to demonstrate a real-time imaging capability.

It turns out that the software being written is ideal for use on other projects, such as other smart sensors and networked control systems. One possibility is a health maintenance network of spacecraft sensors. One factor which helps this extensibility is the fact that the software is DOS-based. PC/104 embedded controllers can run extended DOS. The current software already contains the most important features needed to run the real-time portion (also called the background process) on an embedded controller. What remains is to actually rearrange the software classes such that they will compile and run on both a standalone PC and an embedded controller.

When the background process software migrates to an embedded controller, the component which remains on the console PC, called the foreground process, can evolve into a standard PC application. This enables the author to migrate the foreground classes onto an advanced operating system such as Windows NT, which enables graphics windows and integrated network support. Thus the foreground application can evolve into an application which runs across the Internet, allowing more than one console to operate or monitor a network of sensors. (Use of simple device drivers could allow the background process to run under Windows NT at reduced speed.) The sensors themselves can easily be turned into Internet nodes among other protocols. This is exactly the sort of development needed to create the next generation of spacecraft control system. In addition, the software could be used in commercial applications that require flexibility and high performance.

Thus the goal of producing the first hydrogen leak imaging system is coincident with that of producing the software needed on the next generation of spacecraft. An ideal set of software components would have foreground and background classes which could be used under different combinations of desktop PC's and embedded controllers, and various combinations of operating systems.

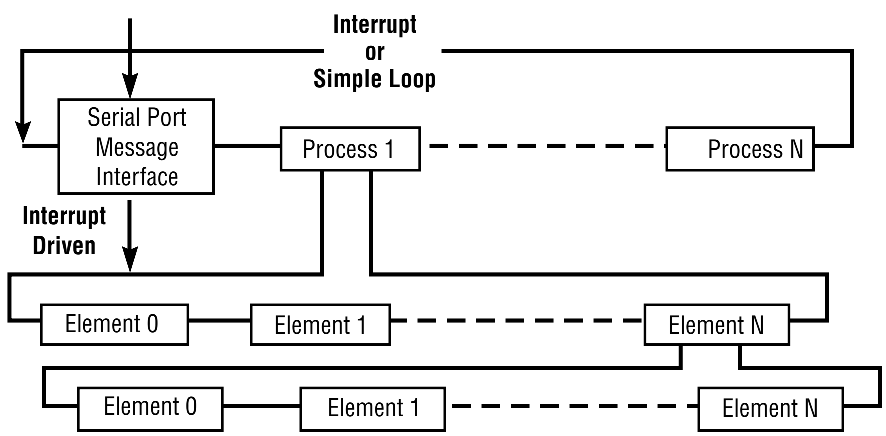

FIGURE 90.—Scheduler for background process (PC/104 system). Scheduler can nest: registry allows all elements (i.e., class instances) to point to one another. Each element can respond to queries via serial port for debugging.

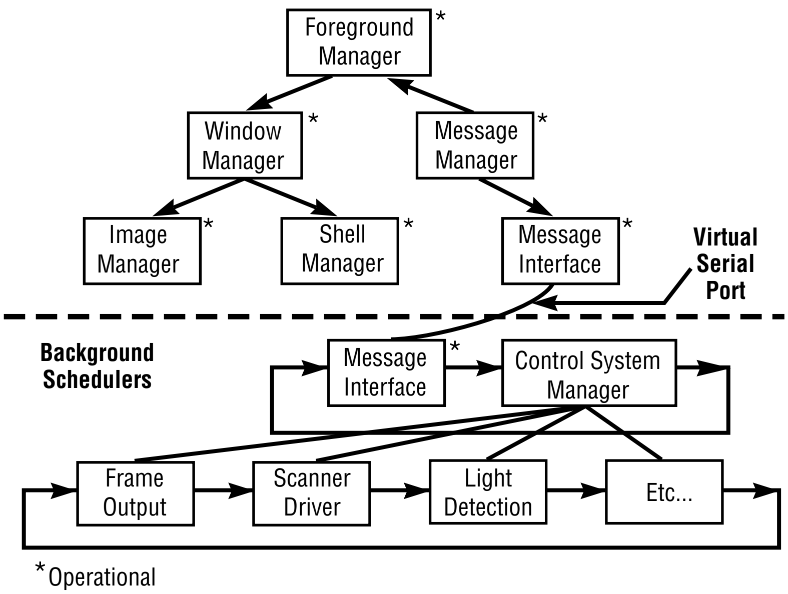

FIGURE 91.—Software architecture.

Sponsor: RLV—Long-Term/High-Payoff Technologies Program

Biographical Sketch: W.T. Powers is a senior measurement systems engineer in the Instrumentation Branch of the Astrionics Laboratory of MSFC. He holds a bachelor of science in electrical engineering, minors in mechanical, nuclear and physics, from Tennessee Polytechnic Institute. Powers has 33 years service with NASA and primarily deals with development of advanced sensors and measurement, acquisition, and processing systems.

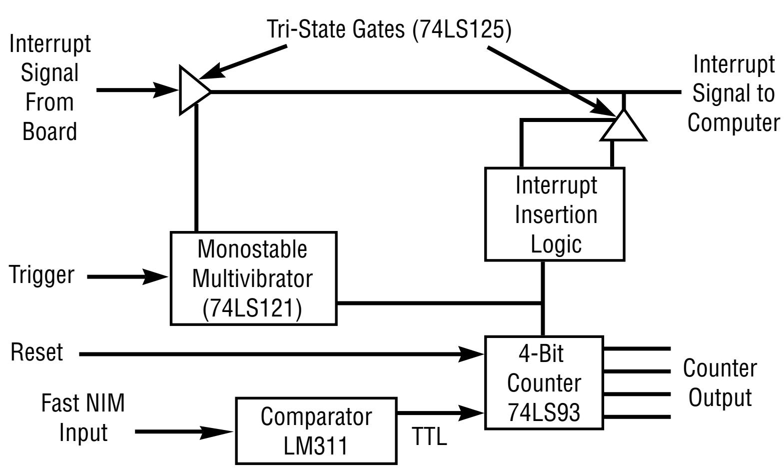

FIGURE 92.—Interface circuit block diagram.